基于旋转基技术的多目标粒子群优化算法 *

戴永彬，纪川川，康伟伟

(辽宁工业大学 软件学院，辽宁 锦州 121001)

摘 要：为了解决多目标优化求解的问题，提出一种基于旋转基技术的多目标粒子群优化算法(rtMOPSO)。首先，改进了旋转基可视化技术并将 Pareto 前沿映射到改进的旋转基扇形平面上，采用差熵指标监测种群进化状态；其次，为平衡归档集的收敛性和多样性，提出了角度支配和角度支配力度两种新的概念并设计归档集新的排序方法；最后，在融合了旋转基角度和距离的概念的基础上，提出了一种改进的全局引导粒子的选择策略。改进算法采用两个类别的测试函数，与 5 种多目标优化算法进行了对比实验。实验结果表明，改进算法在收敛性和多样性方面优势明显。

关键词：粒子群；多目标优化；旋转基技术

中图分类号：TP301.6　　doi: 10.3969/j.issn.1001-3695.2018.02.0138

Multi-objective particle swarm algorithm based on rotation basis

Dai Yongbin, Ji Chuanchuan, Kang Weiwei

(*College of Software Liaoning University of Technology*, *Jinzhou Liaoning* 121001, *China*)

Abstract: This paper proposed a multi-objective particle swarm algorithm based on rotation basis (rtMOPSO) in order to solve problems about multi-objective optimization. Firstly, Pareto front maps to the improved sectorial plane of rotation basis so that evolutionary status of the population were detected by the entropy and its difference entropy of the population. Then, angle dominance and strength of angle dominance were proposed to design an archive maintaining strategy which can balance diversity and convergence. Finally, selecting the global best solution was proposed based on rotation angle and distance of the rotation basis. The improved algorithm is compared with 5 multi-objective optimization algorithms on two kinds of test suites. The Simulation results show that improved algorithm in this paper has big advantages over the other competitors in terms of diversity and convergence.

Key words: particle swarm; multi-objective optimization; rotation basis technique

0　引言

近年来，多目标优化问题受到了学术界和工程界广泛关注，取得了大量的研究成果。然而当多目标优化问题的维数大于 3 时，如何准确呈现 Pareto 前沿的分布状态从而实施相应的控制策略获取最优 Pareto 解集，是一项十分困难的工作。可视化技术是一种解决多目标优化问题的重要手段[1]。可视化技术主要采用降维算法映射 Pareto 前端，分为无损降维和有损降维。无损降维方法实现降维不会损坏数据信息，主要包括平行坐标系、旋转基坐标等方法[2-4]；有损降维首先获取高维数据的特征和关系，将其映射到低维空间，主要包括主成分分析、高斯随机过程模型、聚类等[5,6]。文献[4]利用旋转基可视化技术反映 Pareto 解优劣情况，方便决策者进行分析和决策，但是不能监控粒子种群的进化状态，有效解决多目标优化问题。

粒子群可以快速收敛，但容易早熟，陷入局部最优。虽然可以采取措施避免粒子种群早熟，但是一定会影响粒子种群后期的多样性。因此如何平衡局部收敛性和种群多样性急需有效的解决方案。为此，进一步完善归档集的管理与维护、全局引导粒子的选取以及种群进化状态的监测等方面的算法成为提高粒子群收敛性和多样性的关键所在。

目前，在归档集的管理方面，国内外学者提出了多种维护策略进，主要分为两类：第一类方法是根据种群密度的大小剪裁归档集，例如，自适应网格、拥挤距离、δ 支配[8,9]等方法。第二类方法是依据归档集粒子和参考点或参考区的距离整理归档集，例如，参考区域、参考点等偏好算法 [10,11]。在全局引导粒子的选取方面，现已提出了大量的性能优异的算法和改进算法，如 sigma 以及改进算法、灰色关联、信息熵法、差分算法、动态加权、轮盘赌[12-14]等方法。通过这些算法可以获得全局最优粒子，从而保证了粒子种群的正确飞行方向，提高了种群的收敛性和分布性。另外，还有多种根据种群进化状态调节收敛

收稿日期：2018-02-05；**修回日期：**2018-04-14　　**基金项目：**辽宁省自然科学基金资助项目（2013020036）

作者简介：戴永彬（1972-），男，辽宁绥中人，教授，博士，主要研究方向为智能控制、预测控制、冶金自动化（dyb16@163.com）；纪川川（1991-），男，硕士研究生，主要研究为电力系统自动化；康伟伟（1993-），女，硕士研究生，主要研究为电力系统自动化.

性和多样性的平衡算法，如动态调整惯性权重和学习因子、种群规模自适应算法、种群的交叉与变异算法、多层次信息交互算法等，但是这些算法对种群进化状态还未能准确监测，因此也就不能清楚地辨别进化状态，从而有效地执行进化策略。为了监测种群进化状态，文献[8]提出了状态熵以及差熵的概念，对归档集状态熵的变化情况进行分析和辨识，从而确定粒子群进化状态的边界阈值。但文献[8]使用密度排序方法维护归档集，这种单一排序方法不能同时满足归档集收敛性和多样性的维护需求。

综合以上分析，本文将平行坐标和文献[4]提出的旋转基技术相结合，改进了旋转基映射策略从而实现高维数据到二维空间的映射，使决策者可以对 Pareto 解集的进行准确地分析和决策。其次，在归档集的维护和管理方面提出了旋转基角度支配和角度支配力度的新概念，设计了归档集双排序算法。另外，本文将旋转基角度和距离作为全局最优粒子的选择指标，因此可以获取综合性能最强的粒子作为全局最优粒子。最后，对文献[8]提出的基于 Pareto 熵的状态监测方法进行了改进，通过监测种群状态选择不同的归档集排序方法，实现了归档集的综合动态管理，提高了种群的收敛性和分布性。

1　多目标优化问题和粒子群基本原理

1.1　多目标优化问题

多目标问题可描述如下：

$$\min f(X)=\{f_1(X),f_2(X)\ldots,f_m(X)\}$$
$$s.t.\begin{cases} g_i(x)\ge 0 & i=1,2,\cdots,p \\ h_j(x)=0 & j=1,2,\cdots,q \end{cases} \quad (1)$$

其中：X 为 $\boldsymbol{R}$ 空间的决策变量；$g_i(x)$为和 $h_j(x)$分别是不等式约束和等式约束。

多目标优化问题一般采用 Pareto 解集作为寻优的结果。Pareto 解集内一般有若干非劣解，可以根据决策需要选取合适的非劣解作为最优解，在 Pareto 前沿均匀分布并无限靠近真实值，这是 Pareto 解集的最理想的状态。

1.2　粒子群算法基本原理

粒子群是一种基于群体仿生的进化算法。x 代表粒子的位置，v 代表粒子的速度，两者的关系可以描述为

$$\begin{aligned} v_i^{t+1} &= wv_i^t + c_1r_1(p_i^t - x_i^t) + c_2r_2(g_i^t - x_i^t) \\ x_i^{t+1} &= x_i^t + v_i^{t+1} \end{aligned} \quad (2)$$

其中：$v_i=(v_{i1},v_{i2},\cdots,v_{iD}),x_i=(x_{i1},x_{i2},\cdots,x_{iD})$。另外，$w$ 为惯性权值，个体最优值和全局最优值分别是 $p_i=(p_{i1},p_{i2},\cdots,p_{iD}),g=(g_1,g_2,\cdots,g_D)$，$c_1$、$c_2$ 是学习因子，r_1，r_2 是介于[0,1]的随机数，t 表示粒子群迭代更新次数，D 代表粒子的维数。

2　旋转基技术及进化状态监测的改进

多维数据可视化属于信息可视化的研究领域，主要应用在数据挖掘、网络、社交、文本等可视化方面。旋转基技术是重要可视化技术之一，可以将多维数据直接呈现给用户或决策者，辅助决策。但该算法不适合直接用于种群进化状态的监控，文献[4]也没有将其用于多目标优化问题的求解。为此，本文对旋转基进行了改进并应用于多目标粒子群优化算法。

2.1　旋转基技术的改进

旋转基可视化算法的基本原理是利用旋转基向量的旋转角度表示 Pareto 解在各维目标上的取值情况，再将各目标上的基向量叠加，在 90 扇形平面上形成一条折线。绘制点的旋转角度表示该解的性能，终点向量到原点的长度或距离表示该解性能的波动情况。标准化后的旋转角度公式如下所示：

$$\theta_i^k = \frac{f_i^k - f_i^m}{f_i^M - f_i^m}\times\frac{\pi}{2} \quad (3)$$

其中：i 为目标的数量，i=1,2，…M；k 表示种群的粒子的序号，k=1,2，… K；f^m、f^M 为适应度函数的最大值和最小值。

旋转基向量的连接过程如下：首先，设定一组固定长度的路径向量基{w_i，i=1,2,… M}，向量的数量等于目标的维数。其次，以原点为起点，按照水平方向，逆时针旋转角度 θ_i^k， 得到 Pareto 解的第一个目标的映射坐标，然后再以该点为起始坐标，采用同样方式绘制下一个目标的映射坐标。依次类推即可完成第 M 个目标映射坐标的绘制。

综上分析可知，旋转基可视化技术可以表示 Pareto 解的性能的优劣程度，但是没有采用类似密度等指标展示全部粒子在 Pareto 前沿分布状况[8]。为此，本文将旋转基和平行坐标结合起来，改进旋转基技术。具体如下：根据种群粒子的数量将旋转基 90 度扇形区域划成 K 个扇区，每个扇区的圆心角的角度为 π /2K。然后采用平行坐标的思想，将目标维数取 M 条垂直轴线，每条轴线上的数值范围为目标适应度函数的最小值到最大值并且均匀分布。为了将旋转基融入平行坐标，本文将平行坐标的 M 条垂直轴线转换为旋转基 90 度扇区内的 M 条同心圆圆弧，这样就形成 KxM 个扇块。经过平行坐标映射的旋转基平面称为旋转基扇区平面，被同心圆圆弧所分割的扇区平面称为旋转基扇块。如，令 K=3，M=4，这样形成共计 12 个扇块，具体如图 1 所示。

图 1　旋转基扇区平面

这样，高维的 Pareto 前沿被映射成旋转基扇区平面内，种群粒子也同时被映射到旋转基扇块内。通过分析扇区块内的粒子的数量可以获知 Pareto 前沿粒子的分布状态。为了计算每个粒子的目标分量所在的扇块位置，可用如下公式表示。如第 k

个粒子第 i 个目标分量映射到旋转基扇区的扇块编号 H 为

$$H_i^k = \left\lceil \frac{K \cdot \theta_i^k}{\pi / 2} \right\rceil \quad (4)$$

其中：$\lceil x \rceil$ 表示向上取整数；H_i^k 为[1，K]的整数；如果 θ_i^k 为 0 时，可设 H_i^k 为 1。例如，假定多目标优化问题有 4 个目标，粒子总数为 3，按照公式（4），可计算出每个粒子编号。例如，第 1 个粒子的第 2 个目标分量编号为 3，即 k_1i_3 的 H 值为 3，可写成 H_2^1=3。具体分布如图 2 所示。

图 2　旋转基扇区平面中 Pareto 前沿

2.2　状态熵及进化状态阈值

2.2.1 状态熵

本文采用了 Pareto 熵和差熵[9]监测 Pareto 前沿的进化状态。Pareto 熵和差熵能够准确描述旋转基扇区平面上种群粒子分布情况。

1）状态熵公式

$$E(t) = -\sum_{H=1}^{K}\sum_{i=1}^{M} \frac{Sector_i^H(t)}{KM} \log \frac{Sector_i^H(t)}{KM} \quad (5)$$

其中：$Sector_i^H(t)$ 表示落在第 H 行第 i 列扇块内的坐标分量的数量。

2）差熵的公式

$$\Delta E = E(t) - E(t-1) \quad (6)$$

2.2.2 进化状态阈值的确定

粒子群种群在进化过程中会经历收敛状态、多样性状态和停滞状态等三个状态。其中，多样性状态为种群收敛和停滞的中间状态，因此只需要确定中间状态差熵的最大边界阈值 α 和最小阈值 β，就可以确定进化状态。为此，设定种群分布的一种极端情况：Pareto 最优解集有 3 个粒子且有 3 个目标分量，粒子 k_3 的所在的每个扇块内都有两个分量，具体如图 3 所示。当 Pareto 前沿迭代更新后，出现的第一种极端情况是，旧粒子 k_3 被更新的粒子 k_3 替换并全部落入了 3 个空闲的扇块，具体如图 4 ；另一种情况是更新粒子 k_3 只落入了 1 个空闲的扇块。归档集内发生了更新的粒子，其差熵发生变化，其他没有更新的粒子差熵值不变，因此只要计算出更新前后发生变化的粒子熵的差值即可计算出归档集差熵变化。具体如式（7）（8）所示。

$$\alpha = \left(\sum_{i=1}^{M}\frac{1}{QM}\log\frac{1}{QM} - \sum_{i=1}^{M}\frac{2}{QM}\log\frac{2}{QM}\right) + \left(\sum_{i=1}^{M}\frac{1}{QM}\log\frac{1}{QM} - \sum_{i=1}^{M}\frac{0}{QM}\log\frac{0}{QM}\right) = \frac{2}{Q}\log 2 \quad (7)$$

$$\beta = \frac{2}{MK}\log 2 \quad (8)$$

其中，为 Q 当前归档集 $\boldsymbol{A}$ 中粒子的数量。

因此，当$|\Delta E|>\alpha$ 或 $Q(t)\neq Q(t\text{-}1)$ 时，种群处于收敛状态；当 $A=K$ 且 $\beta<|\Delta E|<\alpha$ 时处于多样化状态；当 $A=K$ 且$\Delta E|<\beta$ 时，种群处于停滞状态。

图 3　更新前的情况

图 4　更新后的情况

3　归档集的维护与全局引导粒子选取

3.1　归档集的维护与管理

粒子群的全部非支配解一般存放在归档集中。一个高效的归档集维护策略是实现归档集收敛性和多样性的保证。目前，多数归档集采用单一的管理策略很难兼顾归档集的收敛性和多样性指标。例如，归档集经常采用粒子密度作为归档集排序的指标[8]，这种策略侧重于选择压力的控制；有的采用自适应网格策略则侧重于分布性的控制。实际上，归档集的维护策略应该根据种群进化状态自适应切换，当种群处于收敛状态时，为了保持收敛速度，保留收敛性强的粒子；当种群处于多样性状态时，为维护归档集的多样性，删除分布密度比较高的粒子。为此，本文采用双排序的维护策略，在不同的进化阶段选择不同排序策略，平衡选择压力和分布性。本文提出了收敛程度指标和分布程序指标作为双排序的依据，具体如下

3.1.1 收敛程度指标

为了度量种群的收敛程度，本文提出了角度支配和角度支配力度的概念并将角度支配力度作为收敛程度的指标。

1）角度支配概念

归档集的任意两个解为 x 和 y，两个解的旋转基角度分别是 θ_x^i,θ_y^i，称 x 角度支配 y 当且仅当

$$\forall i = 1,2,\cdots M, \theta_x^i \le \theta_y^i \wedge \exists j = 1,2,\cdots M, \theta_x^j < \theta_y^j \quad (9)$$

2）角度支配力度概念

归档集的任意一个解 $x\in A$，其支配力度 S_i 是角度支配其他解的数量。

$$Si=\{u|x\in A, x\prec u\} \tag{10}$$

角度支配力度表明了归档集中粒子向 Pareto 前沿收敛的程度，角度支配力度越大，说明该解越靠近真实的 Pareto 前沿。

3.1.2 分布程度指标

$$F_i=\sqrt{\frac{1}{K-1}(\bar{d}-d_i)^2} \tag{11}$$

其中：K 为 Pareto 解集中解的总数；$d_i=\min(\sum_{k=1}^{M}\left|H_k^i(x)-H_k^j(x)\right|)$，（$i\neq j$）表示某个非劣解的目标向量到最优解目标向量的距离；$\bar{d}$ 是 di 的平均值。

分布程度指标 F_i 反应了非劣解集对应前端的均匀程度。其值越小，分布性能越好。

3.1.3 旋转基折线距离

归档集粒子排序时，如果粒子 S_i 和 F_i 的指标都相同，则采用旋转基映射折线距离作为辅助判断指标。旋转基折线距离公式，具体如下：

$$L_k=\frac{1}{M^2}(M+\sum_{i=1}^{M}\sum_{j=1(j\neq i)}^{M}(\cos\theta_i^k-\theta_j^k)) \tag{12}$$

旋转基折线距离越大说明该粒子在各个目标上性能波动越小，性能越均衡。

3.1.4 归档集的排序与维护

根据进化状态，如果种群处于收敛阶段，先按照 S_i 、F_i 和 L_k 次序，对关键字 S_i 按升序而 F_i 和 L_k 按降序排序；如果种群处于多样化阶段，则归档集粒子可按照 F_i、S_i 和 L_k 次序，对对关键字 S_i 按升序而 F_i 和 L_k 按降序排序。最后，按照进化状态情况以及归档集规模情况，将归档集中排序靠后的多余粒子裁剪掉即可。

3.2 全局最优粒子的选取

粒子群全局最优粒子决定着种群进化方向，也影响着种群的选择压力和种群的多样性。因此，为了选取综合性能最强的粒子作为全局最优粒子必须建立一种有效的、合理的选取策略。本文利用旋转基可视化技术的优点，采用旋转角度和旋转基折线距离作为全局最优粒子的选取依据。φ 越小，旋转基折线距离越大，该粒子的多目标综合性能越好，公式如下：

$$\varphi^k=\arctan(\frac{\sum_{i=1}^{M}\cos\theta_i^k}{\sum_{i=1}^{M}\sin\theta_i^k}) \tag{13}$$

全局最优粒子的一般选取过程如下：首先按照旋转角度排序，选择旋转角度最小的粒子作为全局最优粒子；如果有若干粒子的旋转角度相同，则比较其旋转基折线距离，然后选择旋转基折线距离最长的粒子作为全局最优粒子。

根据以上分析可知，在文献[8]算法的基础上，对归档集的维护和全局最优粒子选取进行了改进，使本文算法的性能有较大提升。首先，文献[8]对归档集的维护只采用密度指标进行单排序策略。总所周知，密度指标主要反应归档集粒子的多样性。因此，该策略没有将粒子排序情况和种群的收敛性能结合在一起。本文算法可根据种群的进化状态，采用双排序策略，同时保证了归档集粒子的多样性和收敛性。

表 1　格占优强度与旋转基角度

粒子	f_1	f_2	f_3	L_1	L_2	L_3	格占优强度	旋转基角度
P_1	0.6010	0.0003	0.0047	4	1	1	0	0.4636
P_2	0.5270	0.0309	0.0232	4	4	1	0	1.0242
P_3	0.4715	0.0309	0.0655	3	4	2	0	1.0383
P_4	0.2081	0.0031	0.2002	1	1	4	0	0.5215

其次，在全局最优粒子的选取中，文献[8]采用了格占优强度和密度混合算法。实际上，格占优强度算法在一些特定条件下，区分粒子的优劣能力不足。例如，假设存在 4 个粒子 3 个目标最优解，其格占优强度皆为 0，具体如表 1 所示。为此，必须利用密度指标进行二次排序才能选出全局 P_1。格占优强度算法的分辨力之所以较弱，主要是由于该算最优粒子。本文的旋转基的可视化选取策略，可以直接区分 4 个粒子的优劣程度，一次排序计算即可获知最优粒子为法在算计平行格坐标时采用向上取整的方法，会出现计算误差。平面网格划分的格数越少，损失的信息越多，造成的误差也就越大。相反，本文算法的旋转基角度映射函数会将粒子之间的微小差别放大。另外，计算旋转基角度时没有信息损失，因此对粒子的优劣具有较强的分辨能力，这也是可视化技术的主要特点。

4　rtMOPSO 算法流程与时间复杂度分析

4.1　算法的流程

a)种群参数初始化。粒子群种群主要参数包括种群的归档集规模 K、种群的规模 n、最大循环次数 t_{max}，权重及学习因子 w、c_1、c_2。将归档集粒子数量 k，迭代循环变量 t 等参数清零。

b)进化状态判断。将归档集粒子按照式（3）（4）映射到旋转基扇区平面上；然后利用式（7）（8）计算状态熵和差熵；最后，按照 2.2 节的进化状态的阈值判断进化状态。

c)归档集管理与维护。根据进化状态，按照 3.1 节的规则对归档集粒子排序并剪裁多余粒子。

d)根据 3.2 节策略更新全局最优粒子。

e)按照式（2）更新粒子群种群。

f)如果大于或等于最大循环次数则停止运行，输出最优解集；否则，跳到 b)。

4.2　算法时间复杂度分析

在 rtMOPSO 算法中，为了便于计算，设置粒子群种群和归档集的大小都为 N，多目标优化的目标数量为 M，粒子维数为 D。根据算法的流程，rtMOPSO 算法主要步骤的时间复杂度的分析如下：

Step1，初始化复杂度为 $O(ND)$；Step2，旋转基扇区映射为

线性操作，状态熵和差熵的时间复杂度为 $O(NM)$；在 Step3 中，首先从粒子群种群中选择有效粒子进入归档集，其复杂度为 $O(MN^2)$。其次，执行维护策略时需要分别计算收敛程度、分布程度和辅助判断指标，其时间复杂度为 $O(NM^2)$，多指标排序的复杂度为 $O(N\log N)$；Step5，粒子种群的位置和速度更新为线性操作。综上所述，本文算法的最大时间复杂度的值为 $O(MN^2)$。因此，本文算法和文献[8]算法以及主要采用 Pareto 支配、NSGA 快速排序等方法的多目标粒子群优化算法的时间复杂度相同。

表 2　测试结果

	Problem	NSGAII	SPEA2	MOEAD	dMOPSO	SMPSO	rMOPSO
GD (均值)	DTLZ1	8.0280e-3	6.8079e-2	5.8709e-3	1.6370e+1	8.5590e+0	**5.7273e-3**
	DTLZ2	1.1326e-3	1.0332e-3	**5.0483e-4**	1.5412e-2	3.8664e-3	6.1321e-4
	DTLZ6	4.9899e-6	4.7047e-6	3.6114e-3	1.9039e-2	4.8176e-6	**3.7941e-6**
	ZDT1	3.0491e-5	2.3469e-5	8.6386e-5	6.0845e-4	3.8624e-5	**0.1350e-5**
	ZDT2	1.8500e-5	4.6469e-6	1.8572e-4	5.5901e-4	2.9975e-5	**2.7368e-6**
	ZDT3	3.9224e-5	3.2040e-5	2.1738e-4	4.1661e-4	4.3316e-5	**2.5246e-5**
Spread (均值)	DTLZ1	4.8551e-1	2.7821e-1	**1.6938e-1**	7.9282e-1	8.3284e-1	1.7325e-1
	DTLZ2	5.3215e-1	9.1660e-2	1.7041e-1	2.3881e-1	4.0660e-1	**8.4357e-2**
	DTLZ6	7.3369e-1	1.2879e-1	2.0492e+0	1.9683e+0	4.4293e-1	**1.1258e-1**
	ZDT1	4.4291e-1	1.2359e-1	2.7783e-1	2.7843e-1	4.1422e-1	**1.0139e-1**
	ZDT2	4.4236e-1	1.2982e-1	1.4822e-1	3.2305e-1	4.1902e-1	**1.0846e-1**
	ZDT3	4.5690e-1	1.1906e-1	5.8705e-1	5.6158e-1	3.9600e-1	**0.9937e-1**
HV (均值)	DTLZ1	1.2142e-1	**1.3848e-1**	1.2491e-1	5.3803e-3	1.3352e-2	**1.8945e-1**
	DTLZ2	7.0738e-1	7.3819e-1	**7.4475e-1**	5.5509e-1	6.7827e-1	7.0857e-1
	DTLZ6	1.3270e-1	1.3316e-1	1.2103e-1	1.2121e-1	1.3284e-1	**1.4183e-1**
	ZDT1	8.7091e-1	8.7186e-1	8.7036e-1	8.6267e-1	8.7034e-1	**8.9032e-1**
	ZDT2	5.3771e-1	5.3858e-1	5.3734e-1	4.4506e-1	5.3721e-1	**5.5126e-1**
	ZDT3	1.0202e+0	1.0237e+0	1.0139e+0	1.0101e+0	1.0232e+0	**1.0328e+0**

表 3　部分对比算法的仿真图形

函数	dMOPSO	SMPSO	NSGAII	rtMOPSO
ZDT1				
ZDT2				
ZDT3				

DTLZ1

DTLZ2

DTLZ6

5　实验与分析

5.1　测试算法及对比算法选择

为了检验改进算法rtMOPSO的算法性能，本文选取ZDT[15]系列和DTLZ[16]系列中6个函数作为测试集，包括ZDT1、ZDT2、ZDT3和DTLZ1、DTLZ2、DTLZ6。其中ZDT1、ZDT2、ZDT3三个函数测试改进算法在凸和非凸以及不连续的二维Pareto前沿上寻优的能力；DTLZ1、DTLZ2、DTLZ6三个函数测试对多目标、多变量函数的寻优能力。对比算法选择了 dMOPSO[17]、SMPSO[18]共两种多目标粒子群算法以及NSGA-II[19]、SPEA2[20]、MOEA/D[21]共三种典型的进化算法进行对比测试。

在比较仿真实验中，为保证实验的客观和公平性，所有的测试函数的主要参数设置一致。例如，粒子种群规模为 100、归档集规模为 100，最大评估次数为 30000；ZDT 系列测试的目标维数设置为 2，DTLZ 系列测试的目标维数设置为 3；进化算法的交叉率设为 0.9，变异概率为 0.1。

5.2　评价方法

为了评价对比算法的收敛性和多样性选择以下三种指标：

a)世代距离（generational distance, GD）[22]反映获取的 Pareto 解集靠近真实值的程度。其值越小，说明越接近真实的前沿。

b)多样性指标（spread）[19]主要表明对解集分布均匀情况。其值越小，说明解集的分布性越好。

c)超体积(hypervolume, HV)[23]可以衡量解集覆盖的目标空间区域范围并同时度量了收敛性和多样性的性能。HV 的值越大，说明解集的综合性能越好.

5.3　实验结果的比较与分析

为了便于实验结果，本文将测试结果列表说明，具体如表 2 和 3 所示。其中表 2 是将 rtMOPSO 和各种对比算法运行 30 次后获得实验结果。根据表 2 可知，rtMOPSO 在二维目标的测试实验中，在 GD、Spread 指标上都优于其他对比算法，因此对 2 维目标问题，本文算法表现出较好的收敛性和多样性；对 3 维目标问题，总体上来看，本文算法的收敛性也比较突出，只有 MOEA/D 的 DTLZ1、DTLZ2 的性能和本文算法接近，DTLZ6 测试，本文算法性能与 SPEA2 持平。实验数据表明，对于大部分算法而言，对 DTLZ1 中的测试结果略有不足，但本文算法在 2、3 维目标的测试过程没有明显缺陷。

为了进一步说明本文算法的有效性，可通过算法的对综合性能进行比对。本文采用了 HV 指标，同时从收敛性和多样性进行对比说明。从表 2 可知，虽然 Spea2 和 MOEAD 是比较优秀的算法，具有较好的综合性能，但本文算法 rtMOPSO 在 HV 方面表现更佳，这表明本文提出的算法能够更好的覆盖 Pareto 前沿，获得最优解集。

为了更加直观、清晰地显示本文算法 rtMOPSO 的优越性，在表 3 中，本文给出了 rtMOPSO 和部分对比算法的仿真曲线。根据表 3 可知，其他算法在三个性能指标上存在不同程度的不足之处。例如 dMOPSO、SMPSO 两种粒子群算法在 3 维目标的性能稍显不足，尤其是在 DTLZ1 的测试实验中，解集的分布性较差；NSGAII算法对 ZDT3 和 DTLZ2 的测试中各种性能较差。本文算法的解集无论在 2 维的 ZDT 真实前沿曲线上，还是 3 维的 DTLZ 真实前沿曲面上都非常接近真实解。因此综合考量，本文算法优于 5 种对比算法。

综上所述，本文算法 rtMOPSO 具有较大优势主要原因是本文算法采用了旋转基可视化技术，有利于决策者选择最优粒子，提高了归档集的管理质量并确保选取了全局最优粒子。另

外，在粒子群算法中，采用了进化状态监测策略。在进化过程中判断进化状态后，选择合理的归档集的排序和搜索策略，使种群同时具有较好的收敛性和多样性。

6 结束语

本文改进了旋转基可视化技术，可以清晰呈现归档集粒子 Pareto 前沿的分布和进化状态，有利于决策者选择合适的策略维护归档集和全局最优粒子。另外，借助 Pareto 熵和差熵的概念监控种群进化的情况并根据总群进化的状态自动选择不同的进化管理策略，解决了粒子群进化过程中快速收敛和多样性的不平衡的矛盾。本文利用旋转基可视化技术设计了双排序的档案集维护策略，改进了传统的按密度排序的单一排序方法，使归档集在收敛性和多样性两个方面同时满足要求。最后，使用旋转基角度和距离，提出了全局最优粒子的选择策略。该策略融合旋转基可视化技术的优点，便于选择综合性能最强的粒子，引导种群向 Pareto 前沿靠近。

参考文献：

[1] 孔维健，丁进良，柴天佑. 高维多目标进化算法研究综述 [J]. 控制与决策, 2010, 25 (3): 321-326. (Kong Weijian, Ding Jinliang, Chai Tianyou. Survey on large-dimensional multi-objective evolutionary algorithms [J]. Control and Decision, 2010, 25 (3): 321-326.)

[2] Claessen J H T, Wijk J J. Flexible linked axes for multivariate data visualization [J]. IEEE Trans on Visualization and Computer Graphics, 2011, 17 (12): 2310-2316

[3] Gettinger J, Kiesling E, Stummer C, *et al.* A comparison of representations for discrete multi-criteria decision problems [J]. Decision Support Systems, 2013, 54 (2): 976-985.

[4] 毕晓君，李博，苍岩. 基于交互式决策的旋转基多目标可视化技术 [J]. 计算机辅助设计与图形学学报, 2014, 26 (5): 739-746. (Bi Xiaojun, Li Bo, Cang Yan. Visualization of high dimension multi-objective of rotation basis based on interactive decision making [J]. (Journal of Computer-Aided Design ＆ Computer Graphics, 2014, 26 (5): 739-746.)

[5] Taghavi T, Pimentel A D, Sabeghi M. VMODEX: a novel visualization tool for rapid analysis of heuristic-based multi-objective design space exploration of heterogeneous MPSoC architectures [J]. Simulation Modelling Practice and Theory, 2012, 22 (5): 166-196.

[6] Zio E, Bazzo R. A clustering procedure for reducing the number of representative solutions in the pareto front of multiobjective optimization problems [J]. European Journal of Operational Research, 2011, 210 (3): 624-634.

[7] Coello C A C, Pulido G T, Lechuga M S. Handling multiple objectives with particle swarm optimization [J]. IEEE Trans on Evolutionary Computation, 2004, 8 (3): 256-279.

[8] 胡旺, Gary G Y, 张鑫. 基于 Pareto 熵的多目标粒子群优化算法 [J]. 软件学报, 2014, 25 (5): 1025-1050. (Hu Wang, Gary G Y, Zhang Xin. Multi-objective particle swarm optimization based on pareto entropy [J]. Journal of Software, 2014, 25 (5): 1025-1050.)

[9] Al M N, Andrei P, John M D. MOPSO: MOPSO based on decomposition and dominance with archiving using crowding distance in objective and solution spaces [J]. IEEE Trans on Evolutionary Computation, 2014, 22 (1): 47-77.

[10] 蒲保兴，杨路明，谢东. 嵌入决策者偏好区域的多目标优化算法 [J]. 小型微型计算机系统, 2009, 1 (1): 144−147. (Pu Baoxing, Yang Luming, Xie Dong. Multi-objective optimization algorithm embedded by user preference region [J]. Journal of Chinese Computer Systems, 2009, 1 (1): 144−147.)

[11] Deb K, Sunder J, Rao N U B, *et al.* Reference point based multi-objective optimization using evolutionary algorithms [J]. International Journal of Computational Intelligence Research, 2006, 2 (3): 273−286.

[12] 黄敏，江渝，毛安，等. 基于全局最优位置自适应选取与局部搜索的多目标粒子群优化算法 [J]. 计算机应用, 2014, 34 (4): 1074-1079. (Huang Min, Jiang Yu, Mao An, *et al.* Multi-objective particle swarm optimization algorithm based on global best position adaptive selection and local search [J]. Journal of Computer Applications, 2014, 34 (4): 1074-1079.)

[13] 于繁华，刘寒冰，戴金波. 求解多目标优化问题的灰色粒子群算法 [J]. 计算机应用, 2006, 26 (12): 2950-2952. (Yu Fuanhua, Liu Hanbing, Dai Jinbo. Grey particle swarm algorithm for multi-objective optimization problems [J]. Computer Applications, 2006, 26 (12): 2950-2952.)

[14] 杨景明，马明明，车海军，等. 多目标自适应混沌粒子群优化算法 [J]. 控制与决策, 2015, 30 (12): 2168-2174. (Yang Jingming, Ma Mingming, Che Haijun, *et al.* Multi-objective adaptive chaotic particle swarm optimization algorithm [J]. Control and Decision, 2015, 30 (12): 2168-2174.)

[15] Zitzler E, Deb K, Thiele L. Comparison of multi-objective evolutionary algorithms: Empirical results [J]. IEEE Trans on Evolutionary Computation, 2000, 8 (2): 173-195.

[16] Deb K, Thiele L, Laumanns M, *et al.* Scalable test problems for evolutionary multiobjective optimization [C]. //Evolutionary Multiobjective Optimization. London: Springer, 2005: 105-145.

[17] Nebro A J, Durillo J J, Garcia-Nieto J, Coello C C, *et al.* SMPSO: a new PSO-based metaheuristic for multi-objective optimization [C]// Proc of Computational Intelligence in Multi-Criteria Decision-Making. 2009: 66–73.

[18] Coello Coello C A. A multi-objective particle swarm optimizer based on decomposition [C]// Proc of the 13th Annual Conference on Genetic and Evolutionary Computation. New York: ACM Press, 2011: 69-76.

[19] Deb K, Pratap A, Agarwal S, *et al.* A fast and elitist multiobjective genetic algorithm: NSGA-II [J]. IEEE Trans on Evolutionary Computation, 2002, 6 (2): 182-197.

[20] Zizler E, Laumanns M, Thiele L. SPEA2: Improving the strength Pareto evolutionary algorithm [R]. Lanusanne: Swiss Federal Institute of Technology Computer Engineering and Networks Laboratory, 2001.

[21] Zhang Qingfu, Li Hui. MOEA//D: A multiobjective evolutionary algorithm based on decomposition [J]. IEEE Trans on Evolutionary Computation, 2007, 11 (6): 712-731
[22] Van V D A, Lamont G B. Evolutionary Computation and Convergence to a Pareto Front [C]// Proc of the Late Breaking Papers at the Genetic Programming Conference. Stanford: Stanford University Bookstore, 1998: 221-228.
[23] Beume N, Fonseca C M. On the complexity of computing the hyper-volume indicator [J]. IEEE Trans on Evolutionary Computation, 2009, 13 (5): 1075-1082.